Week 13 R project

Kihiko Peter

2022-06-04

1. Defining the Question.

(a) Specifing the question.

Kira Plastinina is a Russian brand that is sold through a defunct chain of retail stores in Russia, Ukraine, Kazakhstan,Belarus, China, Philippines, and Armenia. The brand's Sales and Marketing team would like to understand their customer's behavior from data that they have collected over the past year. More specifically, they would like to learn the characteristics of customer groups.

##(b) Metrics for success

(c) Understanding the context

(d) Experimental Design

Problem Definition

Set up your work environment

Data Sourcing

Check the Data

Perform Data Cleaning

Perform Exploratory Data Analysis (Univariate, Bivariate & Multivariate)

Implement the Solution

Challenge the Solution

Importing Relevant Libraries

```
##Importing Libraries we need for this Project analysis.

library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.7     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(data.table)
```

```
##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last

## The following object is masked from 'package:purrr':
##
##     transpose
```

```
library(lattice)
library(caret)
```

```
##
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':
##
##     lift
```

```
library(rmarkdown)
library(tinytex)
library(tidyverse)
library(magrittr)
```

```
##
## Attaching package: 'magrittr'

## The following object is masked from 'package:purrr':
##
##     set_names

## The following object is masked from 'package:tidyr':
##
##     extract
```

```
library(warn = -1)
library(RColorBrewer)
library(ggplot2)
library(lattice)
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(mice)
```

```
##
## Attaching package: 'mice'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```
library(DataExplorer)
```

2. Loading of the Dataset

```
# Loading the Dataset

customer_df = fread('http://bit.ly/EcommerceCustomersDataset')
```

3. Previewing of the dataset

```
# Lets preview the top entries of our dataset

head(customer_df)
```

```
##    Administrative Administrative_Duration Informational Informational_Duration
## 1:              0                       0             0                      0
## 2:              0                       0             0                      0
## 3:              0                      -1             0                     -1
## 4:              0                       0             0                      0
## 5:              0                       0             0                      0
## 6:              0                       0             0                      0
##    ProductRelated ProductRelated_Duration BounceRates ExitRates PageValues
## 1:              1                0.000000  0.20000000 0.2000000          0
## 2:              2               64.000000  0.00000000 0.1000000          0
```

```
## 3:              1               -1.000000  0.20000000 0.2000000          0
## 4:              2                2.666667  0.05000000 0.1400000          0
## 5:             10              627.500000  0.02000000 0.0500000          0
## 6:             19              154.216667  0.01578947 0.0245614          0
##    SpecialDay Month OperatingSystems Browser Region TrafficType
## 1:          0   Feb                1       1      1           1
## 2:          0   Feb                2       2      1           2
## 3:          0   Feb                4       1      9           3
## 4:          0   Feb                3       2      2           4
## 5:          0   Feb                3       3      1           4
## 6:          0   Feb                2       2      1           3
##          VisitorType Weekend Revenue
## 1: Returning_Visitor   FALSE   FALSE
## 2: Returning_Visitor   FALSE   FALSE
## 3: Returning_Visitor   FALSE   FALSE
## 4: Returning_Visitor   FALSE   FALSE
## 5: Returning_Visitor    TRUE   FALSE
## 6: Returning_Visitor   FALSE   FALSE
```

```
# Lets preview the last six entries of the dataset
tail(customer_df)
```

```
##    Administrative Administrative_Duration Informational Informational_Duration
## 1:              0                      0             1                     0
## 2:              3                    145             0                     0
## 3:              0                      0             0                     0
## 4:              0                      0             0                     0
## 5:              4                     75             0                     0
## 6:              0                      0             0                     0
##    ProductRelated ProductRelated_Duration BounceRates  ExitRates PageValues
## 1:             16                503.000 0.000000000 0.03764706    0.00000
## 2:             53               1783.792 0.007142857 0.02903061   12.24172
## 3:              5                465.750 0.000000000 0.02133333    0.00000
## 4:              6                184.250 0.083333333 0.08666667    0.00000
## 5:             15                346.000 0.000000000 0.02105263    0.00000
## 6:              3                 21.250 0.000000000 0.06666667    0.00000
##    SpecialDay Month OperatingSystems Browser Region TrafficType
## 1:          0   Nov                2       2      1           1
## 2:          0   Dec                4       6      1           1
## 3:          0   Nov                3       2      1           8
## 4:          0   Nov                3       2      1          13
## 5:          0   Nov                2       2      3          11
## 6:          0   Nov                3       2      1           2
##          VisitorType Weekend Revenue
## 1: Returning_Visitor   FALSE   FALSE
## 2: Returning_Visitor    TRUE   FALSE
## 3: Returning_Visitor    TRUE   FALSE
## 4: Returning_Visitor    TRUE   FALSE
## 5: Returning_Visitor   FALSE   FALSE
## 6:       New_Visitor    TRUE   FALSE
```

```
# Lets check the shape of the dataset

dim(customer_df)
```

```
## [1] 12330    18
```

```
# The dataset contains 12330 rows and 18 columns
# There are various data types: double, character and logical.
# Some variables need to be encoded.
```

```
# Lets check the datatypes of the data

sapply(customer_df,class)
```

```
##          Administrative Administrative_Duration           Informational
##               "integer"               "numeric"               "integer"
##  Informational_Duration         ProductRelated ProductRelated_Duration
##               "numeric"               "integer"               "numeric"
##             BounceRates               ExitRates              PageValues
##               "numeric"               "numeric"               "numeric"
##              SpecialDay                   Month        OperatingSystems
##               "numeric"             "character"               "integer"
##                 Browser                  Region             TrafficType
##               "integer"               "integer"               "integer"
##             VisitorType                 Weekend                 Revenue
##             "character"               "logical"               "logical"
```

```
# Lets check the unique values

sapply(customer_df,function(x) length(unique(x)))
```

```
##          Administrative Administrative_Duration           Informational
##                      28                    3337                      18
##  Informational_Duration         ProductRelated ProductRelated_Duration
##                    1260                     312                    9553
##             BounceRates               ExitRates              PageValues
##                    1873                    4778                    2704
##              SpecialDay                   Month        OperatingSystems
##                       6                      10                       8
##                 Browser                  Region             TrafficType
##                      13                       9                      20
##             VisitorType                 Weekend                 Revenue
##                       3                       2                       2
```

```
# Lets check the Summary Statistics of the dataset
summary(customer_df)
```

```
##  Administrative   Administrative_Duration Informational
##  Min.   : 0.000   Min.   :  -1.00         Min.   : 0.000
##  1st Qu.: 0.000   1st Qu.:   0.00         1st Qu.: 0.000
##  Median : 1.000   Median :   8.00         Median : 0.000
##  Mean   : 2.318   Mean   :  80.91         Mean   : 0.504
##  3rd Qu.: 4.000   3rd Qu.:  93.50         3rd Qu.: 0.000
##  Max.   :27.000   Max.   :3398.75         Max.   :24.000
##  NA's   :14       NA's   :14              NA's   :14
##  Informational_Duration ProductRelated   ProductRelated_Duration
```

```
##  Min.   :  -1.00        Min.   :  0.00   Min.   :   -1.0
##  1st Qu.:   0.00        1st Qu.:  7.00   1st Qu.:  185.0
##  Median :   0.00        Median : 18.00   Median :  599.8
##  Mean   :  34.51        Mean   : 31.76   Mean   : 1196.0
##  3rd Qu.:   0.00        3rd Qu.: 38.00   3rd Qu.: 1466.5
##  Max.   :2549.38        Max.   :705.00   Max.   :63973.5
##  NA's   :14             NA's   :14       NA's   :14
##   BounceRates         ExitRates         PageValues        SpecialDay
##  Min.   :0.000000   Min.   :0.00000   Min.   :  0.000   Min.   :0.00000
##  1st Qu.:0.000000   1st Qu.:0.01429   1st Qu.:  0.000   1st Qu.:0.00000
##  Median :0.003119   Median :0.02512   Median :  0.000   Median :0.00000
##  Mean   :0.022152   Mean   :0.04300   Mean   :  5.889   Mean   :0.06143
##  3rd Qu.:0.016684   3rd Qu.:0.05000   3rd Qu.:  0.000   3rd Qu.:0.00000
##  Max.   :0.200000   Max.   :0.20000   Max.   :361.764   Max.   :1.00000
##  NA's   :14         NA's   :14
##     Month           OperatingSystems    Browser           Region
##  Length:12330       Min.   :1.000    Min.   : 1.000   Min.   :1.000
##  Class :character   1st Qu.:2.000    1st Qu.: 2.000   1st Qu.:1.000
##  Mode  :character   Median :2.000    Median : 2.000   Median :3.000
##                     Mean   :2.124    Mean   : 2.357   Mean   :3.147
##                     3rd Qu.:3.000    3rd Qu.: 2.000   3rd Qu.:4.000
##                     Max.   :8.000    Max.   :13.000   Max.   :9.000
##
##   TrafficType    VisitorType         Weekend         Revenue
##  Min.   : 1.00   Length:12330       Mode :logical   Mode :logical
##  1st Qu.: 2.00   Class :character   FALSE:9462      FALSE:10422
##  Median : 2.00   Mode  :character   TRUE :2868      TRUE :1908
##  Mean   : 4.07
##  3rd Qu.: 4.00
##  Max.   :20.00
##
```

4. Data Cleaning

```
# Lets check for the number missing values

colSums(is.na(customer_df))
```

```
##          Administrative Administrative_Duration           Informational
##                      14                      14                      14
##  Informational_Duration          ProductRelated ProductRelated_Duration
##                      14                      14                      14
##             BounceRates             ExitRates              PageValues
##                      14                      14                       0
##              SpecialDay                   Month        OperatingSystems
##                       0                       0                       0
##                 Browser                  Region             TrafficType
##                       0                       0                       0
##             VisitorType                 Weekend                 Revenue
##                       0                       0                       0
```

```
# Five columns contained 14 missing values each
```

```
# Lets check for duplicated values in the dataset

duplicates <- customer_df[duplicated(customer_df),]
dim(duplicates)
```

```
## [1] 119  18
```

```
# The dataset contains missing values
```

```
# Finding the percentage of missing values
sum(is.na(customer_df))/prod(dim(customer_df)) * 100
```

```
## [1] 0.05046409
```

```
# Lets deal with the missing values
customer_df  = customer_df %>%

    mutate(Administrative =replace(Administrative,is.na(Administrative),mean(Administrative,na.rm=TRUE))
    mutate(Administrative_Duration =replace(Administrative_Duration,is.na(Administrative_Duration),mean
    mutate(Informational = replace(Informational, is.na(Informational), mean(Informational, na.rm = TRU
    mutate(Informational_Duration =replace(Informational_Duration,is.na(Informational_Duration),mean(In
    mutate(ProductRelated =replace(ProductRelated,is.na(ProductRelated),mean(ProductRelated,na.rm=TRUE))
    mutate(ProductRelated_Duration = replace(ProductRelated_Duration, is.na(ProductRelated_Duration), me
    mutate(BounceRates =replace(BounceRates, is.na(BounceRates),mean(BounceRates,na.rm=TRUE)))%>%
    mutate(ExitRates = replace(ExitRates, is.na(ExitRates), mean(ExitRates, na.rm = TRUE)))
```

```
# Lets check if we have missing values anymore

colSums(is.na(customer_df))
```

```
##          Administrative Administrative_Duration           Informational
##                       0                       0                       0
##  Informational_Duration          ProductRelated ProductRelated_Duration
##                       0                       0                       0
##             BounceRates               ExitRates              PageValues
##                       0                       0                       0
##              SpecialDay                   Month        OperatingSystems
##                       0                       0                       0
##                 Browser                  Region             TrafficType
##                       0                       0                       0
##             VisitorType                 Weekend                 Revenue
##                       0                       0                       0
```

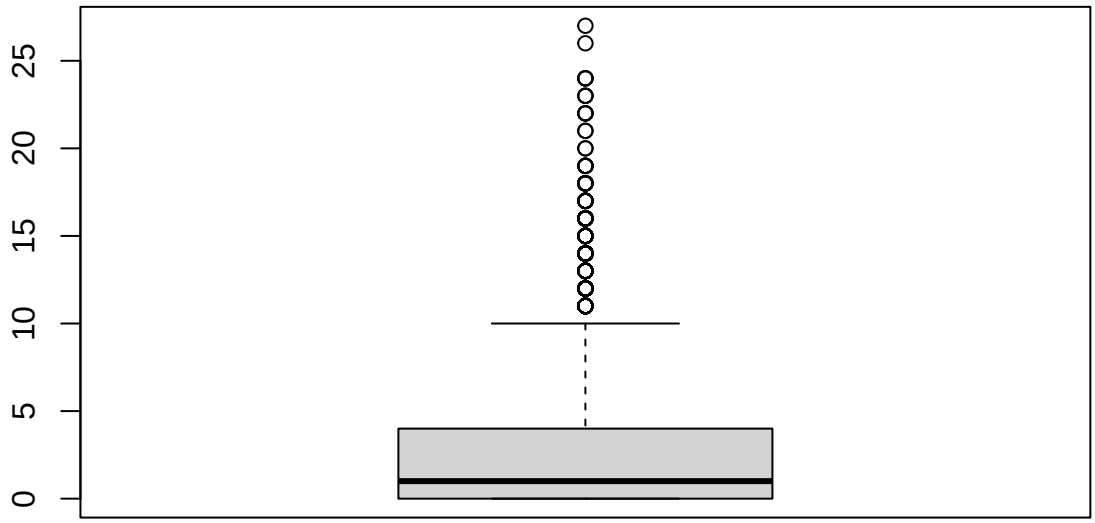
```
# Lets deal with the duplicates by removing them

customer_df <- customer_df[!duplicated(customer_df),]
dim(customer_df)
```

```
## [1] 12211    18
```

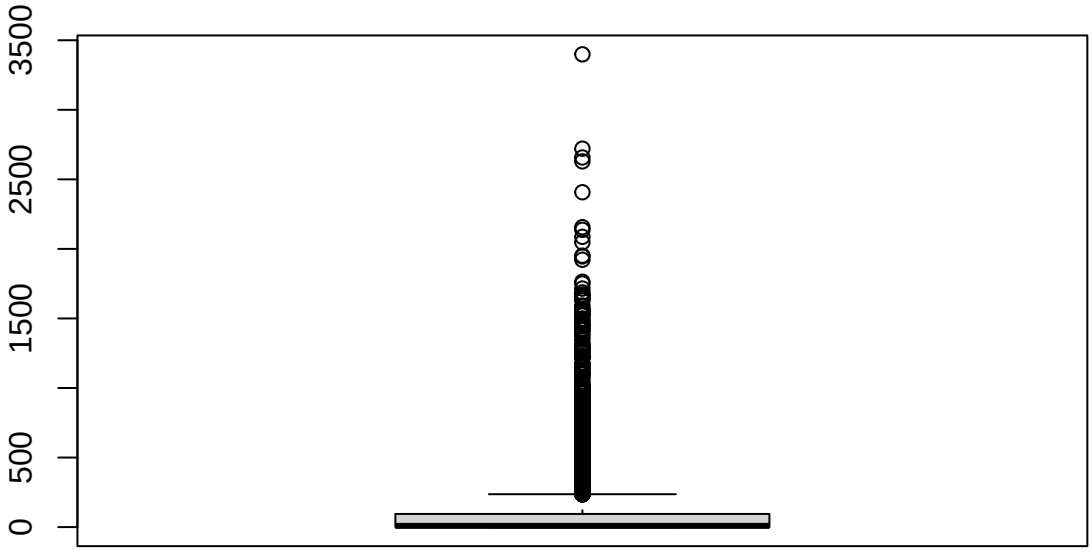
```r
# Lets check if the duplicated values have been removed
duplicates <- customer_df[duplicated(customer_df),]
dim(duplicates)
```

```
## [1]  0 18
```

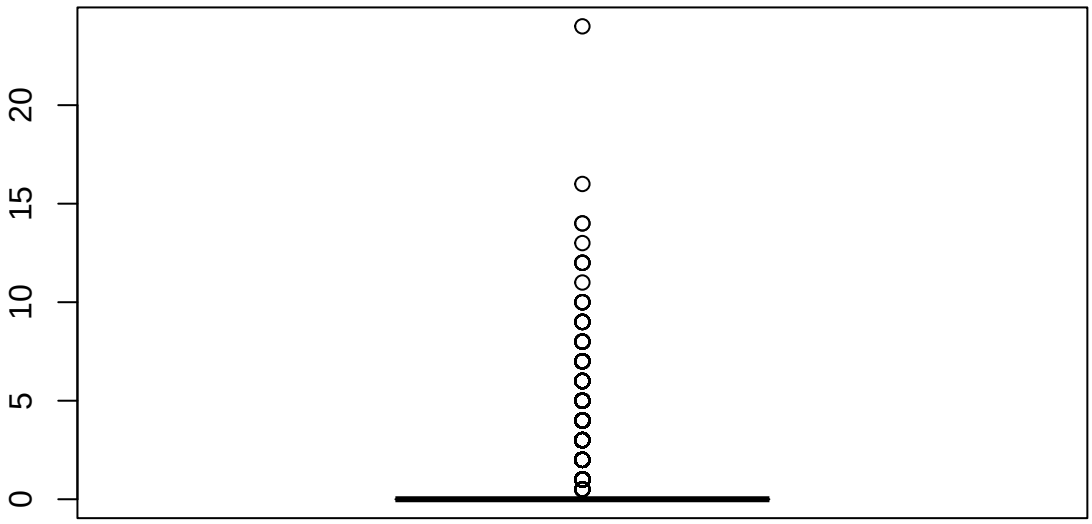
```r
# Lets check for outliers using boxplots

boxplot(customer_df$Administrative)
```



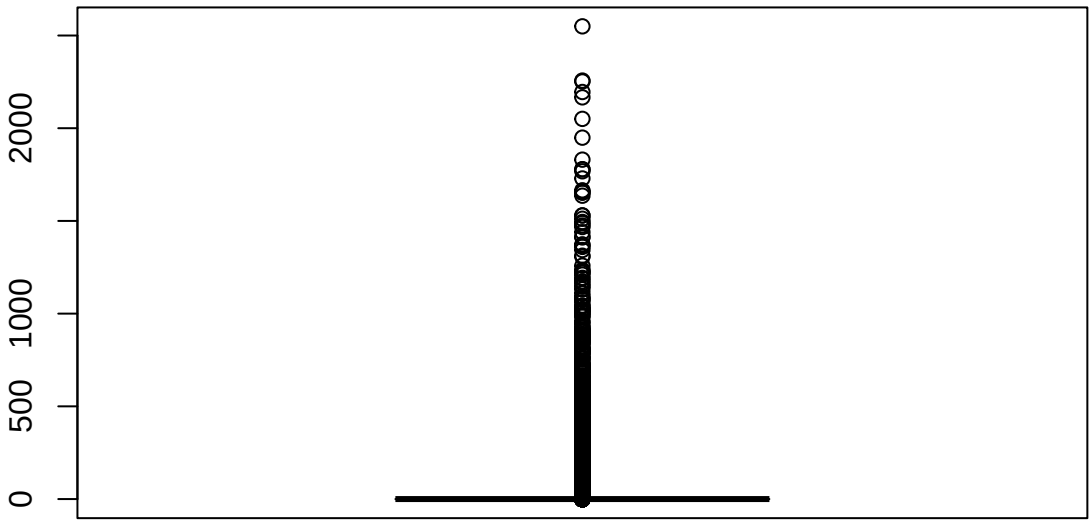

```r
boxplot(customer_df$Administrative_Duration)
```



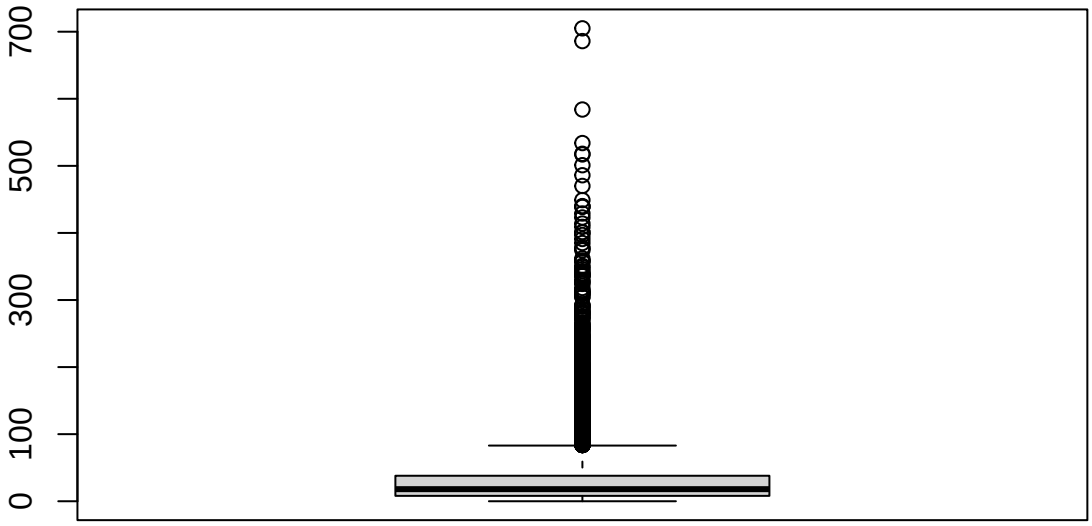

```
boxplot(customer_df$Informational)
```



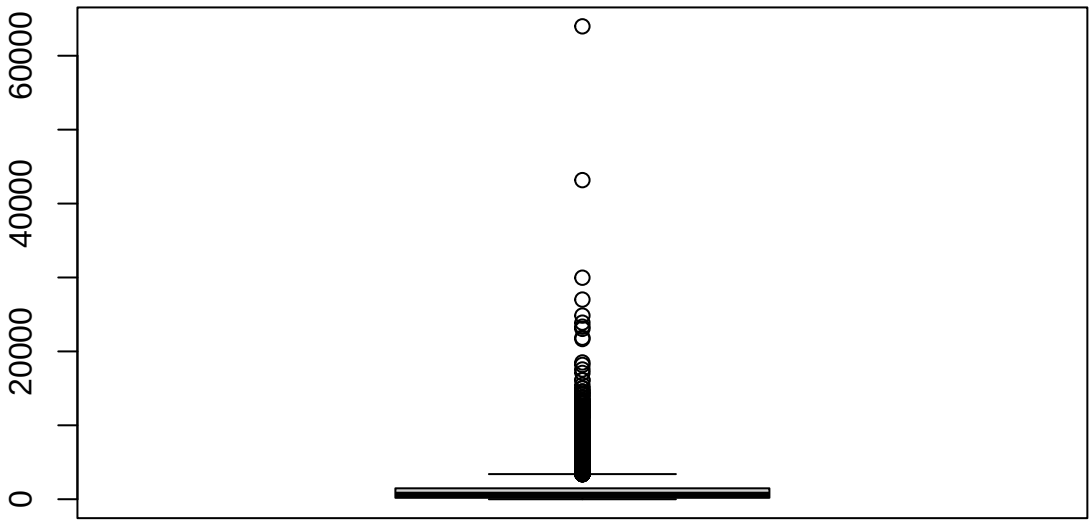

```
boxplot(customer_df$Informational_Duration)
```



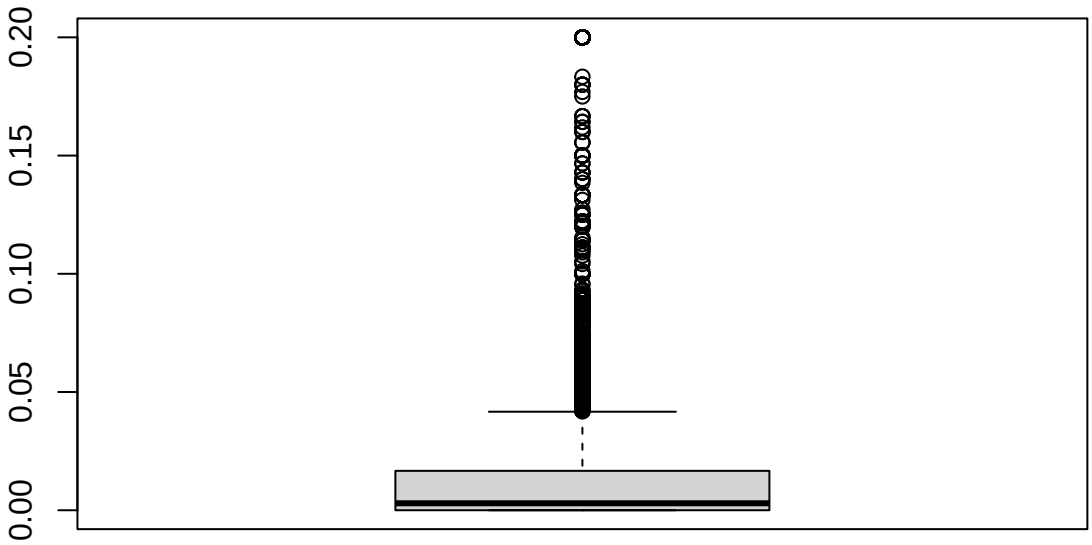

```
boxplot(customer_df$ProductRelated)
```



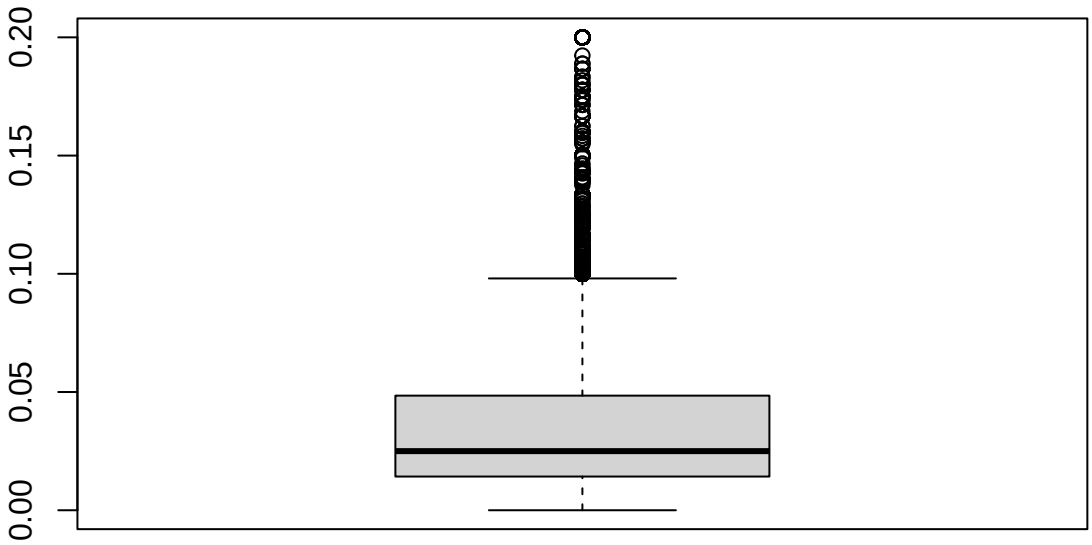

```
boxplot(customer_df$ProductRelated_Duration)
```




```
boxplot(customer_df$BounceRates)
```



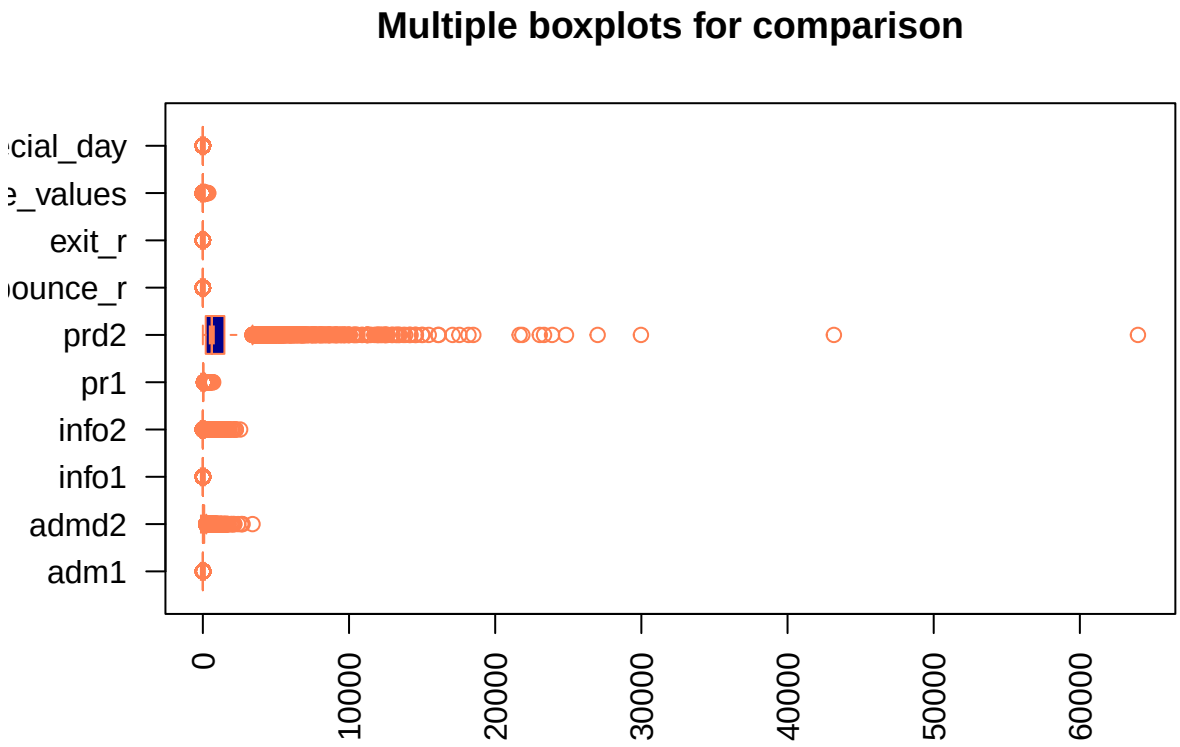

```
boxplot(customer_df$ExitRates)
```




```
# Our dataset contains outliers but let's not remove them since they may be true values
```

```
# Using a boxplot to check for observations far away from other data points.
# Using all double type columns: specifying each
# labeling the title
# labeling the x axis
# specifying color options

adm1 <- customer_df$Administrative
admd2 <-customer_df$Administrative_Duration
info1 <- customer_df$Informational
info2 <- customer_df$Informational_Duration
pr1 <- customer_df$ProductRelated
prd2 <- customer_df$ProductRelated_Duration
bounce_r <-customer_df$BounceRates
exit_r <- customer_df$ExitRates
page_values <-customer_df$PageValues
special_day <- customer_df$SpecialDay



boxplot(adm1, admd2, info1, info2, pr1, prd2, bounce_r, exit_r, page_values, special_day,
main = "Multiple boxplots for comparison",
at = c(1,2,3,4,5,6,7,8,9,10),
names = c("adm1", "admd2", "info1", "info2", "pr1", "prd2", "bounce_r", "exit_r", "page_values", "specia
las = 2,
```

```
col = c("orange","red","blue", "grey", "chartreuse", "blue4", "purple", "green", "gold", "azure"),
border = "coral",
horizontal = TRUE,
notch = TRUE
)
```

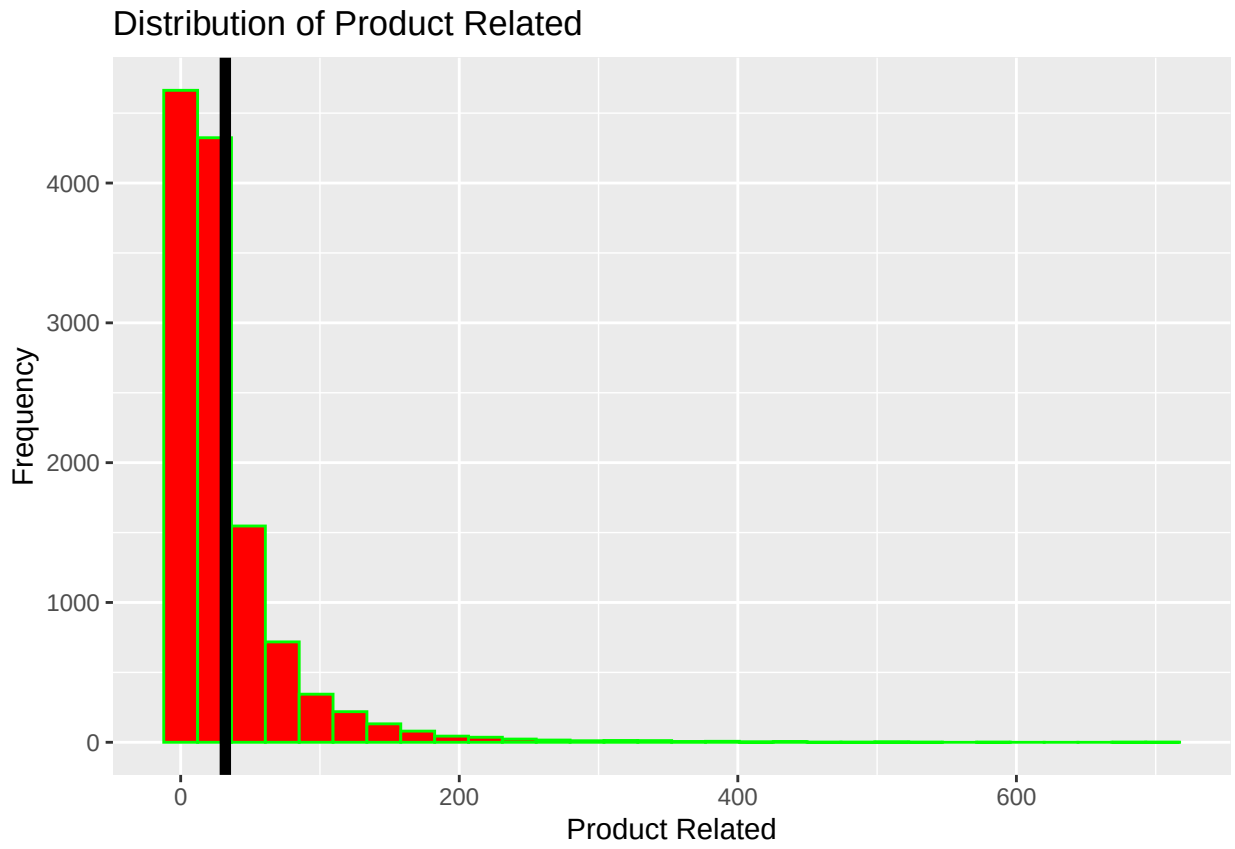


5. Exploratory Data Analysis

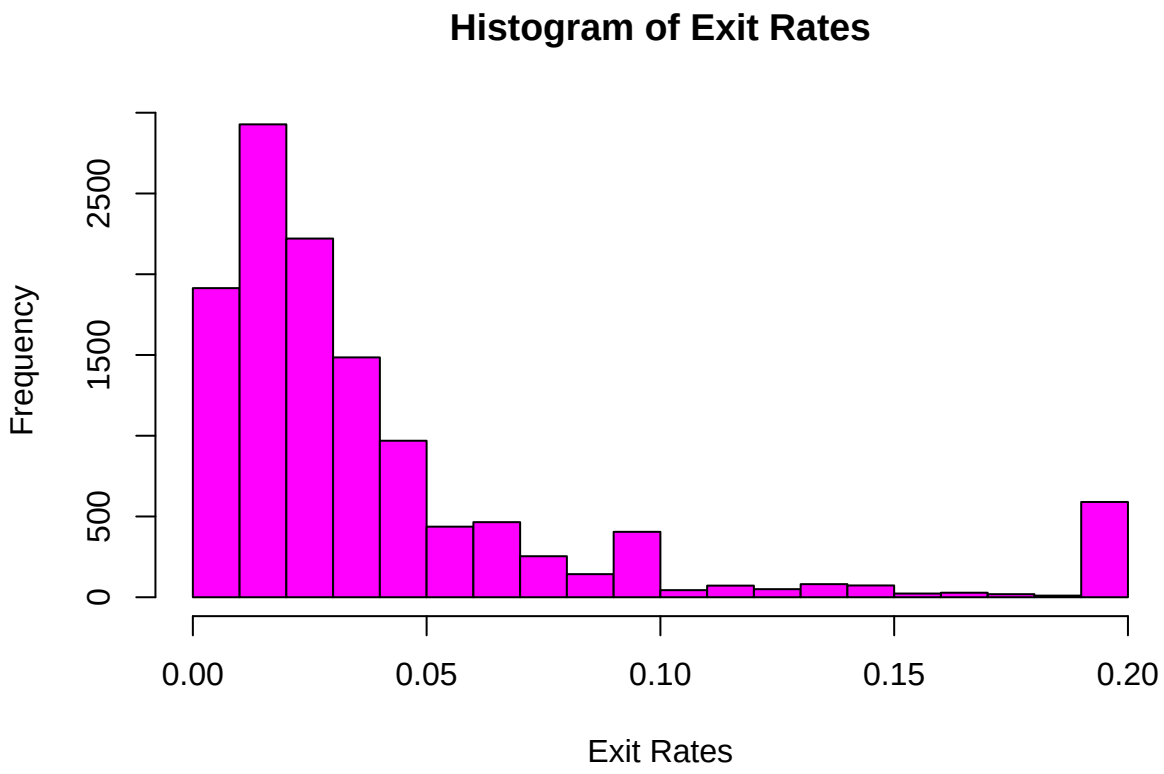
(a) Univariate Analysis

```
# Lets plot a histogram using ggplots
#
#

customer_df %>%
    ggplot(aes(ProductRelated)) +
    geom_histogram(color = "Green",fill = "red") +
    geom_vline(xintercept = mean(customer_df$ProductRelated), lwd = 2) +
    labs(title = "Distribution of Product Related",
         x = "Product Related",
         y = "Frequency")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

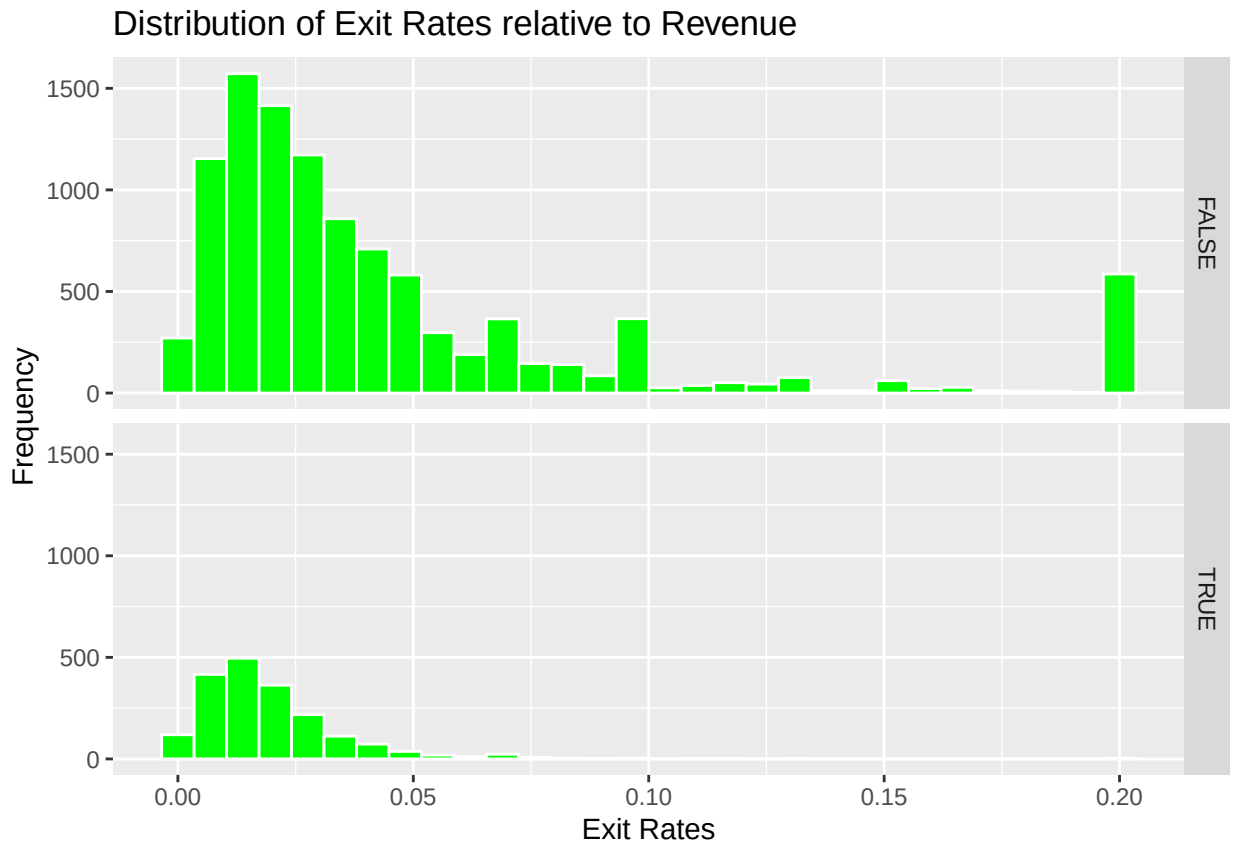



```
# Lets plot a histogram of
hist(customer_df$ExitRates,
     main = "Histogram of Exit Rates",
     xlab = "Exit Rates",
     col = "magenta")
```



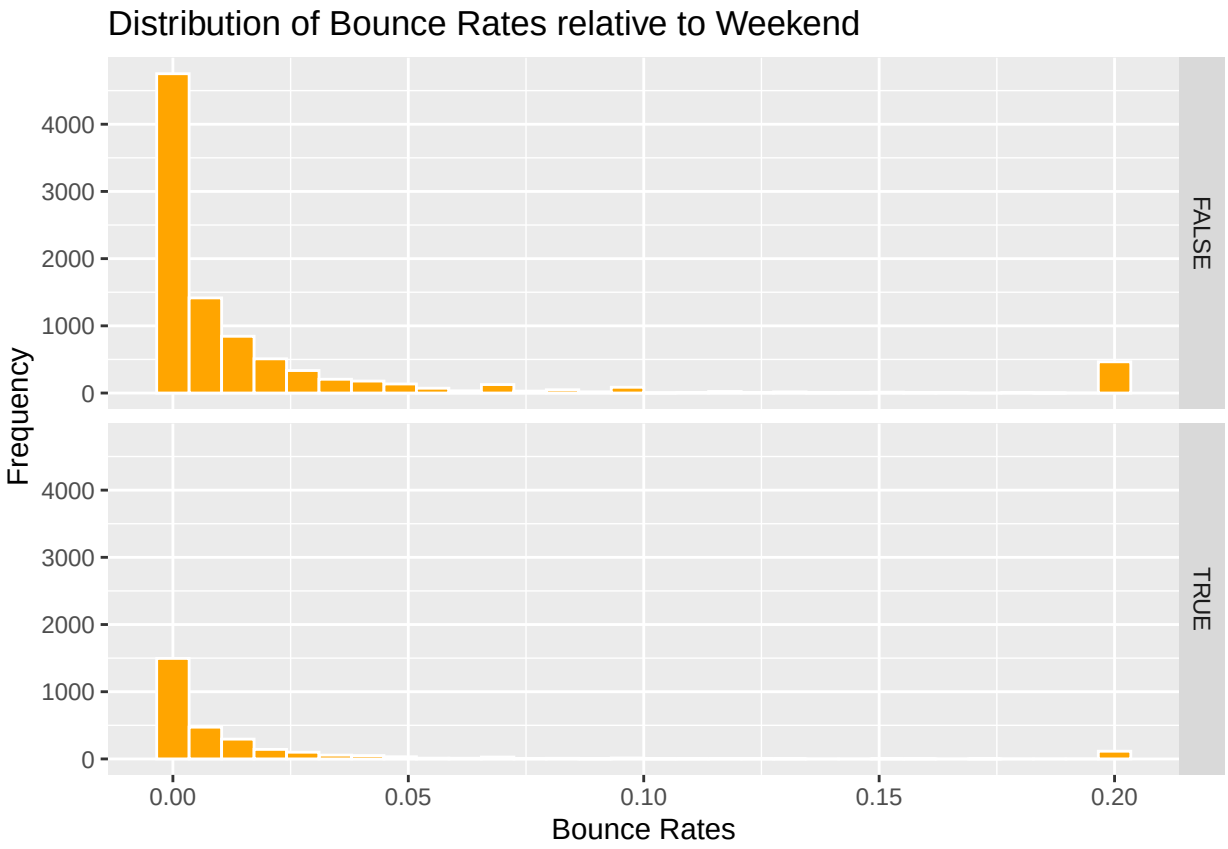

```
# Plotting a histogram Exit Rates relaive to Revenue

customer_df %>%
    ggplot(aes(ExitRates)) +
    geom_histogram(color = "white",fill = "green") +
    labs(title = "Distribution of Exit Rates relative to Revenue",
         x = "Exit Rates",
         y = "Frequency") +
    facet_grid(Revenue~.)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

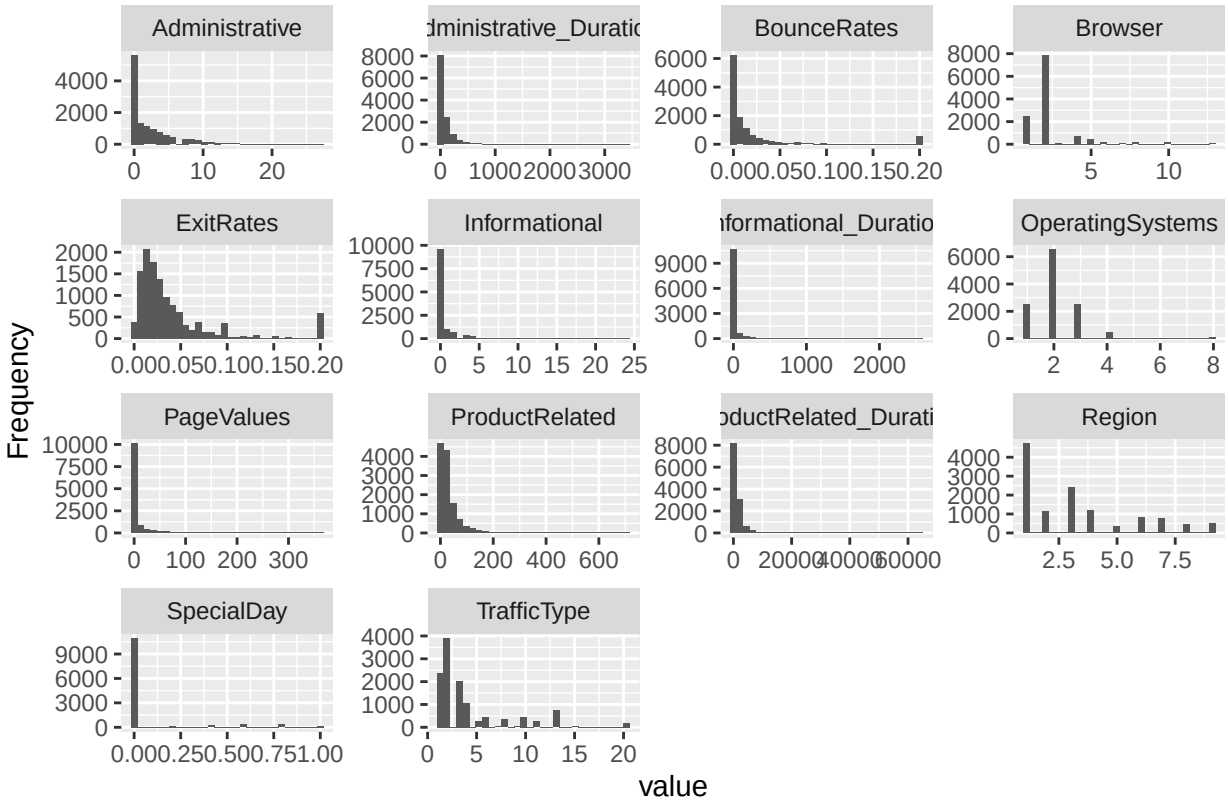



```
# Plotting a Histogram of the Bounce rates relaive to Weekend

customer_df %>%
    ggplot(aes(BounceRates)) +
    geom_histogram(color = "white",fill = "orange") +
    labs(title = "Distribution of Bounce Rates relative to Weekend",
         x = "Bounce Rates",
         y = "Frequency") +
    facet_grid(Weekend~.)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

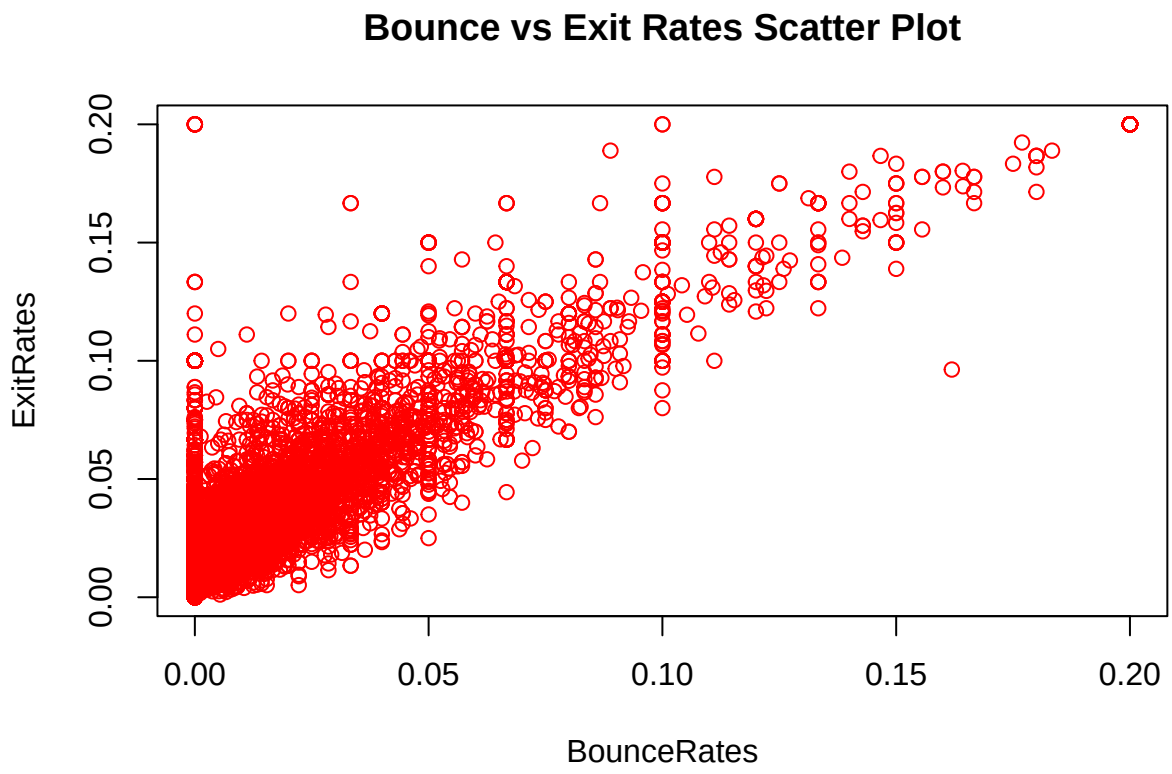



```
# Plotting all histograms in the continuous variables in our data

plot_histogram(customer_df)
```



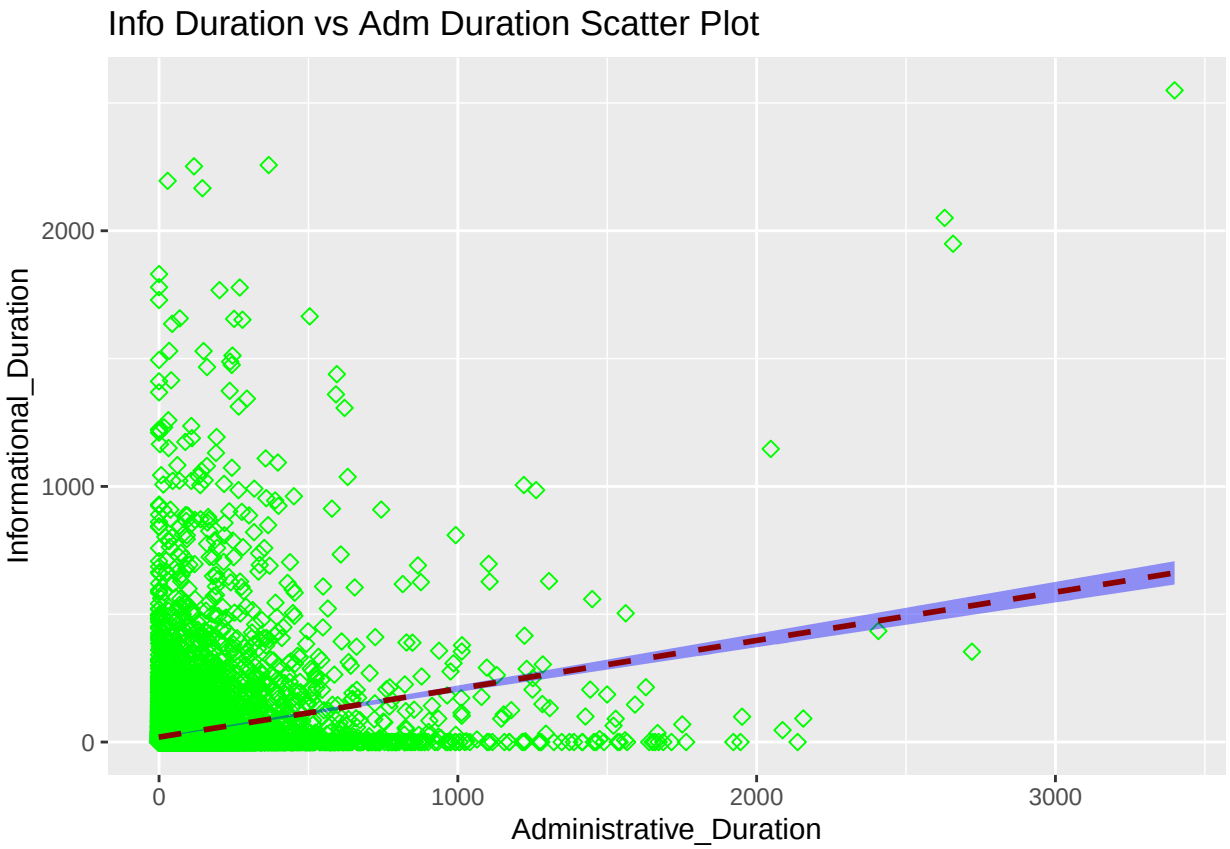
Bivariate Analysis

```
# Plotting a scatter plot using the plot() method

plot(ExitRates ~ BounceRates, data = customer_df,
      col = "red",
      main = "Bounce vs Exit Rates Scatter Plot")
```



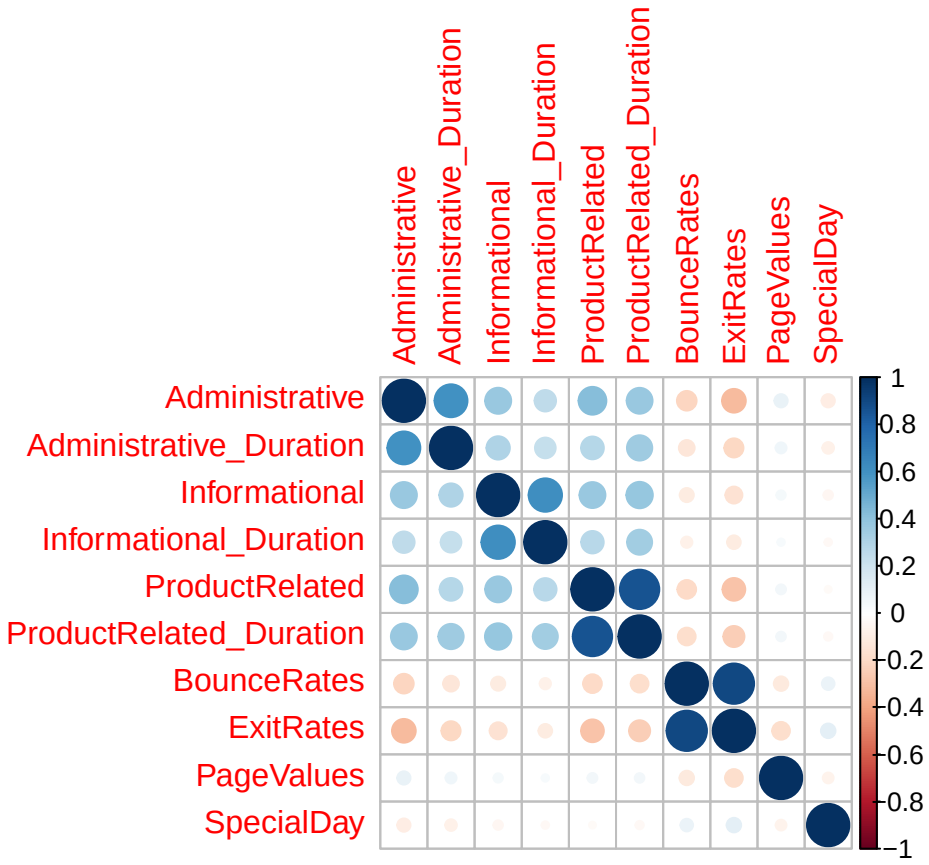

```
# Scatter plot using ggplots and fitting a line of best fit

ggplot(customer_df, aes(x = Administrative_Duration, y = Informational_Duration)) +
       geom_point(size = 2, color= "green", shape = 23)+
       geom_smooth(method=lm,  linetype="dashed",color="darkred", fill="blue")+
       labs(title = "Info Duration vs Adm Duration Scatter Plot")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

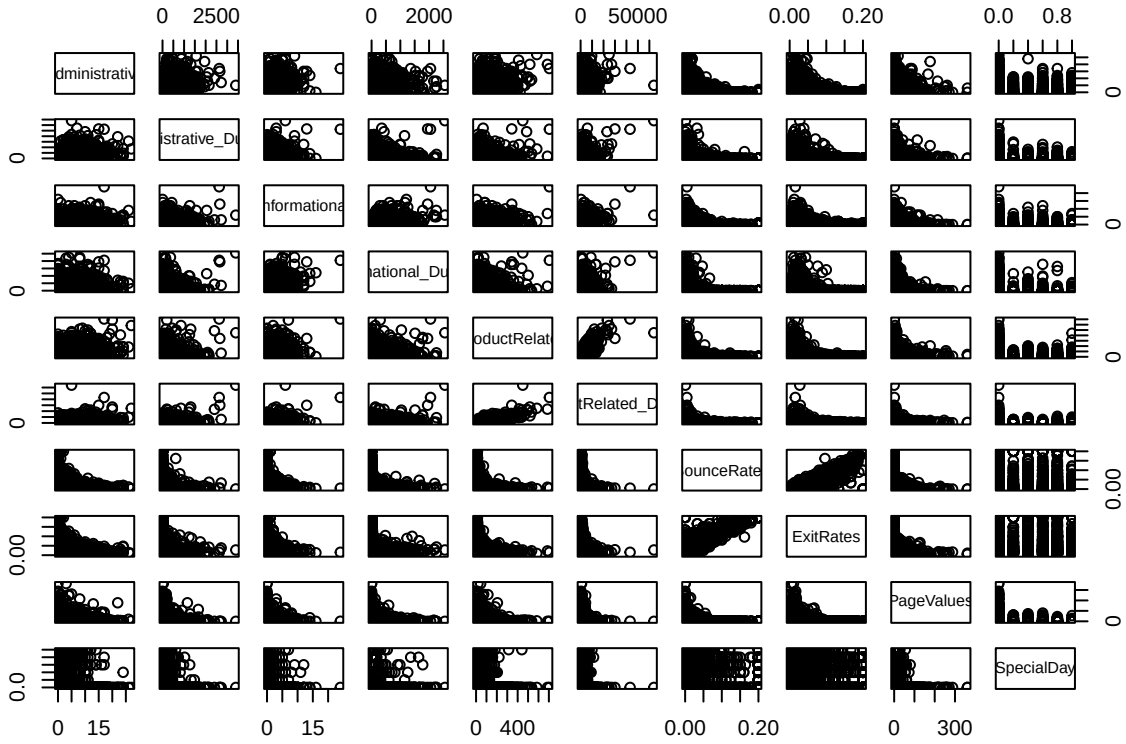


Multivariate Analysis

```
# calculate correlations
correlations <- cor(customer_df[,1:10])
# create correlation plot
corrplot(correlations, method= 'circle')
```




```
# Continuous variables pair plot

pairs(customer_df[,1:10])
```



6. Implementing the Solution

K-means clustering

```
brand.new<- customer_df[, c(1:17)]
brand.class<- customer_df[, "Revenue"]
head(brand.new)
```

```
##    Administrative Administrative_Duration Informational Informational_Duration
## 1:              0                       0             0                      0
## 2:              0                       0             0                      0
## 3:              0                      -1             0                     -1
## 4:              0                       0             0                      0
## 5:              0                       0             0                      0
## 6:              0                       0             0                      0
##    ProductRelated ProductRelated_Duration BounceRates ExitRates PageValues
## 1:              1                0.000000  0.20000000 0.2000000          0
## 2:              2               64.000000  0.00000000 0.1000000          0
## 3:              1               -1.000000  0.20000000 0.2000000          0
## 4:              2                2.666667  0.05000000 0.1400000          0
## 5:             10              627.500000  0.02000000 0.0500000          0
## 6:             19              154.216667  0.01578947 0.0245614          0
```

```
##    SpecialDay Month OperatingSystems Browser Region TrafficType
## 1:          0   Feb                1       1      1           1
## 2:          0   Feb                2       2      1           2
## 3:          0   Feb                4       1      9           3
## 4:          0   Feb                3       2      2           4
## 5:          0   Feb                3       3      1           4
## 6:          0   Feb                2       2      1           3
##          VisitorType Weekend
## 1: Returning_Visitor   FALSE
## 2: Returning_Visitor   FALSE
## 3: Returning_Visitor   FALSE
## 4: Returning_Visitor   FALSE
## 5: Returning_Visitor    TRUE
## 6: Returning_Visitor   FALSE
```

```
# Previewing the class column
# ---
#
head(brand.class)
```

```
##    Revenue
## 1:   FALSE
## 2:   FALSE
## 3:   FALSE
## 4:   FALSE
## 5:   FALSE
## 6:   FALSE
```

```
# converting weekend variables to numerical using dummies
# option max print is set as 99999 to ensure no variable is omitted when het are converted to dummies

brand.new$Weekend = factor(brand.new$Weekend,levels = c('FALSE', 'TRUE'),labels = c(0, 1),options(max.p
head(brand.new$Weekend)
```

```
## [1] 0 0 0 0 1 0
## Levels: 0 1
```

```
dummy <- dummyVars("~ .", "Month + Operating_Systems + Browser + Region +
                   Traffic_Type + Visitor_Type + Weekend", data=brand.new)
encoded <- data.frame(predict(dummy, newdata = brand.new))
brand.new <- cbind(brand.new[ , 1:10], encoded)
```

```
head(brand.new)
```

```
##    Administrative Administrative_Duration Informational Informational_Duration
## 1:              0                       0             0                      0
## 2:              0                       0             0                      0
## 3:              0                      -1             0                     -1
## 4:              0                       0             0                      0
## 5:              0                       0             0                      0
## 6:              0                       0             0                      0
##    ProductRelated ProductRelated_Duration BounceRates ExitRates PageValues
```

```
## 1:              1               0.000000  0.20000000 0.2000000          0
## 2:              2              64.000000  0.00000000 0.1000000          0
## 3:              1              -1.000000  0.20000000 0.2000000          0
## 4:              2               2.666667  0.05000000 0.1400000          0
## 5:             10             627.500000  0.02000000 0.0500000          0
## 6:             19             154.216667  0.01578947 0.0245614          0
##    SpecialDay Administrative Administrative_Duration Informational
## 1:          0              0                      0             0
## 2:          0              0                      0             0
## 3:          0              0                     -1             0
## 4:          0              0                      0             0
## 5:          0              0                      0             0
## 6:          0              0                      0             0
##    Informational_Duration ProductRelated ProductRelated_Duration BounceRates
## 1:                      0              1                0.000000  0.20000000
## 2:                      0              2               64.000000  0.00000000
## 3:                     -1              1               -1.000000  0.20000000
## 4:                      0              2                2.666667  0.05000000
## 5:                      0             10              627.500000  0.02000000
## 6:                      0             19              154.216667  0.01578947
##    ExitRates PageValues SpecialDay MonthAug MonthDec MonthFeb MonthJul
## 1: 0.2000000          0          0        0        0        1        0
## 2: 0.1000000          0          0        0        0        1        0
## 3: 0.2000000          0          0        0        0        1        0
## 4: 0.1400000          0          0        0        0        1        0
## 5: 0.0500000          0          0        0        0        1        0
## 6: 0.0245614          0          0        0        0        1        0
##    MonthJune MonthMar MonthMay MonthNov MonthOct MonthSep OperatingSystems
## 1:         0        0        0        0        0        0                1
## 2:         0        0        0        0        0        0                2
## 3:         0        0        0        0        0        0                4
## 4:         0        0        0        0        0        0                3
## 5:         0        0        0        0        0        0                3
## 6:         0        0        0        0        0        0                2
##    Browser Region TrafficType VisitorTypeNew_Visitor VisitorTypeOther
## 1:       1      1           1                      0                0
## 2:       2      1           2                      0                0
## 3:       1      9           3                      0                0
## 4:       2      2           4                      0                0
## 5:       3      1           4                      0                0
## 6:       2      1           3                      0                0
##    VisitorTypeReturning_Visitor
## 1:                            1
## 2:                            1
## 3:                            1
## 4:                            1
## 5:                            1
## 6:                            1
##    WeekendMonth...Operating_Systems...Browser...Region.....................Traffic_Type...Visitor_T
## 1:
## 2:
## 3:
## 4:
## 5:
```

```
## 6:
##    WeekendMonth...Operating_Systems...Browser...Region.....................Traffic_Type...Visitor_T
## 1:
## 2:
## 3:
## 4:
## 5:
## 6:
```

```
brand.new <- scale(brand.new)
```

```
# Normalizing the a copy of the original data

brand.new <- as.data.frame(apply(brand.new, 2, function(x) (x - min(x))/(max(x)-min(x))))
```

```
head(brand.new)
```

```
##   Administrative Administrative_Duration Informational Informational_Duration
## 1              0            0.0002941393             0           0.0003920992
## 2              0            0.0002941393             0           0.0003920992
## 3              0            0.0000000000             0           0.0000000000
## 4              0            0.0002941393             0           0.0003920992
## 5              0            0.0002941393             0           0.0003920992
## 6              0            0.0002941393             0           0.0003920992
##   ProductRelated ProductRelated_Duration BounceRates ExitRates PageValues
## 1    0.001418440            1.563122e-05  1.00000000  1.000000          0
## 2    0.002836879            1.016029e-03  0.00000000  0.500000          0
## 3    0.001418440            0.000000e+00  1.00000000  1.000000          0
## 4    0.002836879            5.731448e-05  0.25000000  0.700000          0
## 5    0.014184397            9.824223e-03  0.10000000  0.250000          0
## 6    0.026950355            2.426226e-03  0.07894737  0.122807          0
##   SpecialDay Administrative Administrative_Duration Informational
## 1          0              0            0.0002941393             0
## 2          0              0            0.0002941393             0
## 3          0              0            0.0000000000             0
## 4          0              0            0.0002941393             0
## 5          0              0            0.0002941393             0
## 6          0              0            0.0002941393             0
##   Informational_Duration ProductRelated ProductRelated_Duration BounceRates
## 1           0.0003920992    0.001418440            1.563122e-05  1.00000000
## 2           0.0003920992    0.002836879            1.016029e-03  0.00000000
## 3           0.0000000000    0.001418440            0.000000e+00  1.00000000
## 4           0.0003920992    0.002836879            5.731448e-05  0.25000000
## 5           0.0003920992    0.014184397            9.824223e-03  0.10000000
## 6           0.0003920992    0.026950355            2.426226e-03  0.07894737
##   ExitRates PageValues SpecialDay MonthAug MonthDec MonthFeb MonthJul MonthJune
## 1  1.000000          0          0        0        0        1        0         0
## 2  0.500000          0          0        0        0        1        0         0
## 3  1.000000          0          0        0        0        1        0         0
## 4  0.700000          0          0        0        0        1        0         0
## 5  0.250000          0          0        0        0        1        0         0
## 6  0.122807          0          0        0        0        1        0         0
##   MonthMar MonthMay MonthNov MonthOct MonthSep OperatingSystems    Browser
```

```
## 1        0        0        0        0        0        0.0000000 0.00000000
## 2        0        0        0        0        0        0.1428571 0.08333333
## 3        0        0        0        0        0        0.4285714 0.00000000
## 4        0        0        0        0        0        0.2857143 0.08333333
## 5        0        0        0        0        0        0.2857143 0.16666667
## 6        0        0        0        0        0        0.1428571 0.08333333
##   Region TrafficType VisitorTypeNew_Visitor VisitorTypeOther
## 1  0.000  0.00000000                      0                0
## 2  0.000  0.05263158                      0                0
## 3  1.000  0.10526316                      0                0
## 4  0.125  0.15789474                      0                0
## 5  0.000  0.15789474                      0                0
## 6  0.000  0.10526316                      0                0
##   VisitorTypeReturning_Visitor
## 1                            1
## 2                            1
## 3                            1
## 4                            1
## 5                            1
## 6                            1
##   WeekendMonth...Operating_Systems...Browser...Region.....................Traffic_Type...Visitor_Ty
## 1
## 2
## 3
## 4
## 5
## 6
##   WeekendMonth...Operating_Systems...Browser...Region.....................Traffic_Type...Visitor_Ty
## 1
## 2
## 3
## 4
## 5
## 6
```

```
# Applying the K-means clustering algorithm with no. of centroids(k)=3
# ---
#
result<- kmeans(brand.new,2)

# Previewing the no. of records in each cluster
#
result$size
```

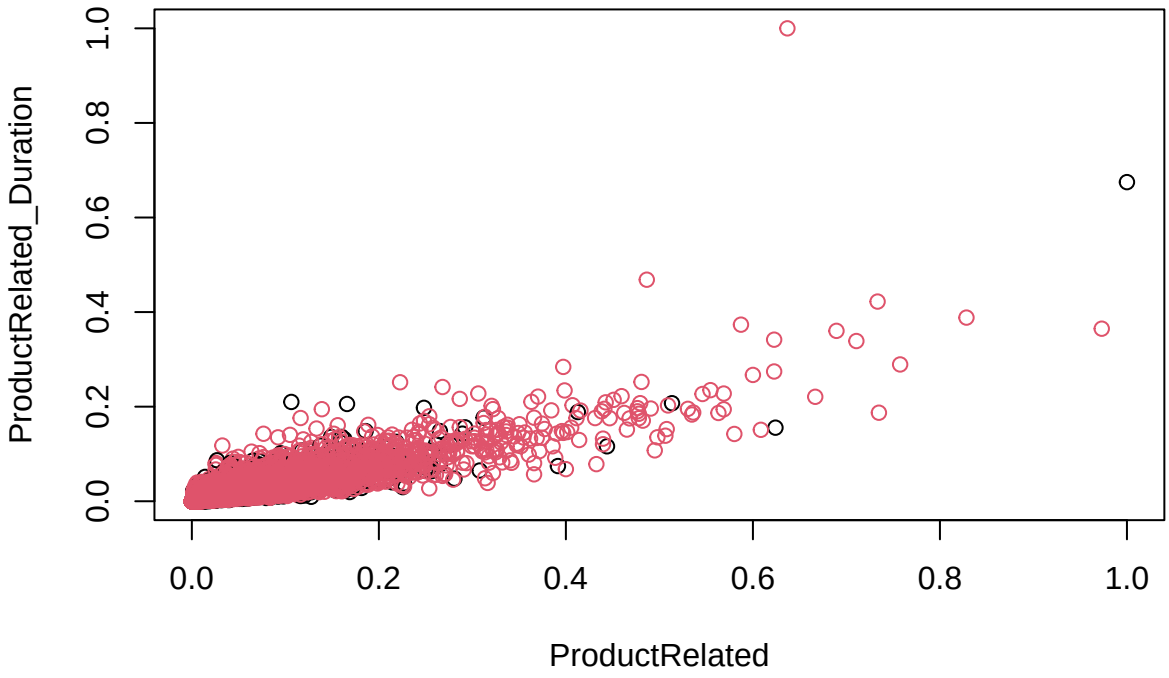
```
## [1] 3331 8880
```

```
#Getting the value of cluster center datapoint value(2 centers for k=2)

centres <-result$centers
head(centres)
```

```
##   Administrative Administrative_Duration Informational Informational_Duration
## 1     0.07352166              0.02093581    0.01784379             0.01115162
```

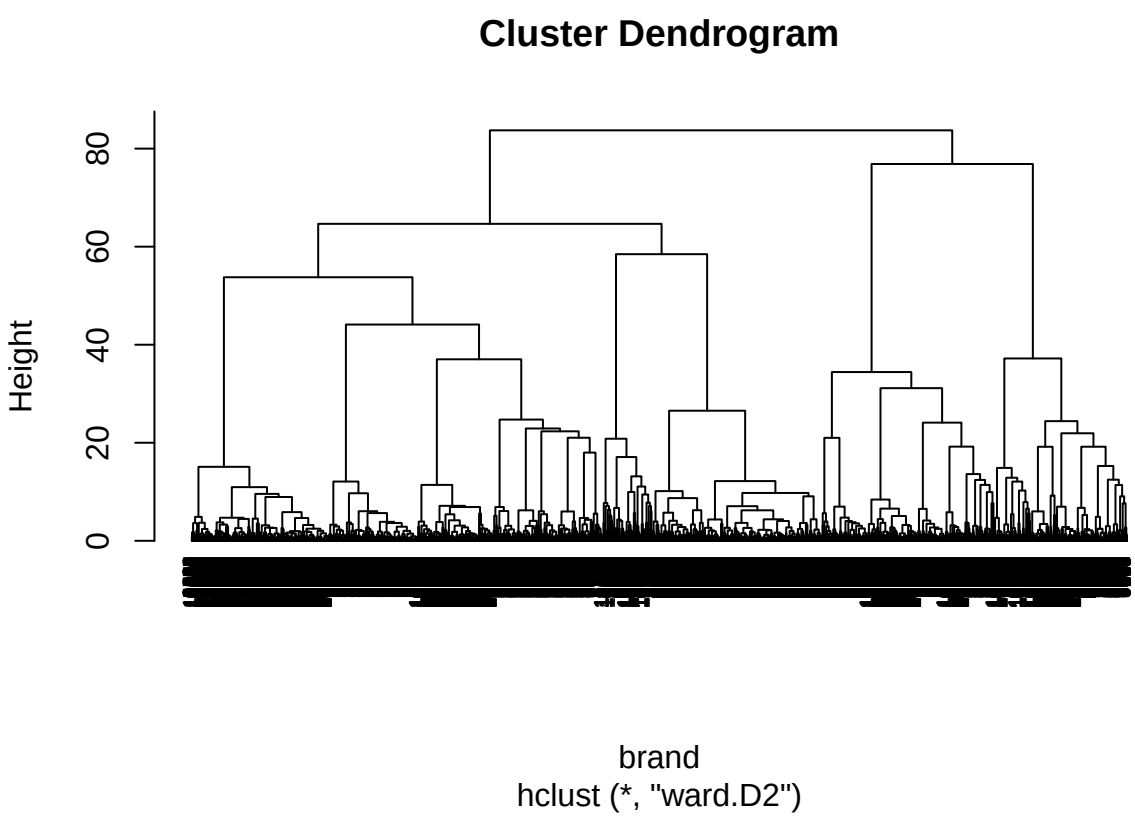
```
## 2     0.09159783              0.02558923    0.02245938             0.01513951
##   ProductRelated ProductRelated_Duration BounceRates ExitRates PageValues
## 1     0.03794007              0.01552144  0.12564008 0.2365999 0.01516289
## 2     0.04829810              0.02015398  0.09346516 0.1965723 0.01691623
##   SpecialDay Administrative Administrative_Duration Informational
## 1 0.21459021     0.07352166              0.02093581    0.01784379
## 2 0.00463964     0.09159783              0.02558923    0.02245938
##   Informational_Duration ProductRelated ProductRelated_Duration BounceRates
## 1             0.01115162     0.03794007              0.01552144  0.12564008
## 2             0.01513951     0.04829810              0.02015398  0.09346516
##   ExitRates PageValues SpecialDay   MonthAug  MonthDec     MonthFeb   MonthJul
## 1 0.2365999 0.01516289 0.21459021 0.00000000 0.0000000 0.0006004203 0.00000000
## 2 0.1965723 0.01691623 0.00463964 0.04876126 0.1921171 0.0202702703 0.04864865
##    MonthJune  MonthMar  MonthMay  MonthNov   MonthOct   MonthSep
## 1 0.00000000 0.0000000 0.9993996 0.0000000 0.00000000 0.00000000
## 2 0.03209459 0.2099099 0.0000000 0.3359234 0.06182432 0.05045045
##   OperatingSystems   Browser    Region TrafficType VisitorTypeNew_Visitor
## 1        0.1604409 0.1145302 0.2677124   0.1831598             0.09576704
## 2        0.1606660 0.1126220 0.2696509   0.1537755             0.15472973
##   VisitorTypeOther VisitorTypeReturning_Visitor
## 1      0.000000000                   0.9042330
## 2      0.009121622                   0.8361486
##   WeekendMonth...Operating_Systems...Browser...Region.....................Traffic_Type...Visitor_Ty
## 1
## 2
##   WeekendMonth...Operating_Systems...Browser...Region.....................Traffic_Type...Visitor_Ty
## 1
## 2
```

```
# Plotting two variables to see how their data points
# have been distributed in the cluster
# Product Related, vs Product Related Duration

plot(brand.new[, 5:6], col = result$cluster)
```



Challenging the solution

```
# We use R function hclust()
# For hierarchical clustering
# First we use the dist() to compute the Euclidean distance btwn obs
# d will be the first argument in the hclust() dissimilairty matrix
#

brand <- dist(brand.new, method = "euclidean")

# We then apply hierarchical clustering using the Ward's method

res.hc <- hclust(brand, method = "ward.D2")

# Lastly we plot the obtained dendrogram
#--

plot(res.hc, cex = 0.6, hang = -1)
```


Cluster Dendrogram
Height
0
20
40
60
80
brand
hclust (*, "ward.D2")